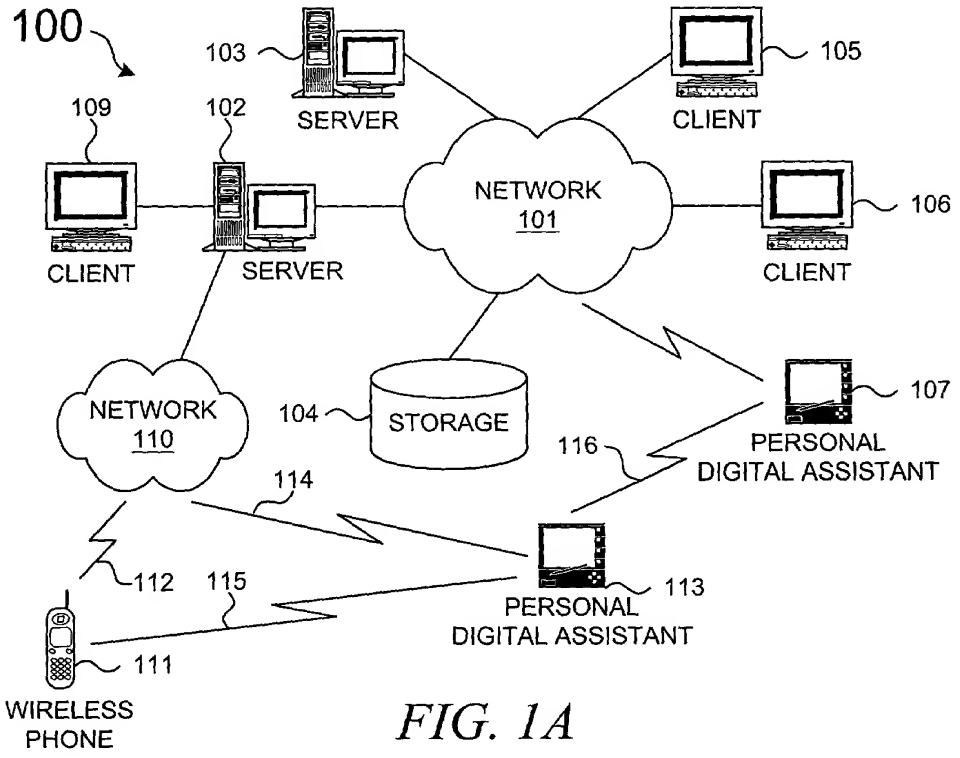

FIG. 1A
(PRIOR ART)

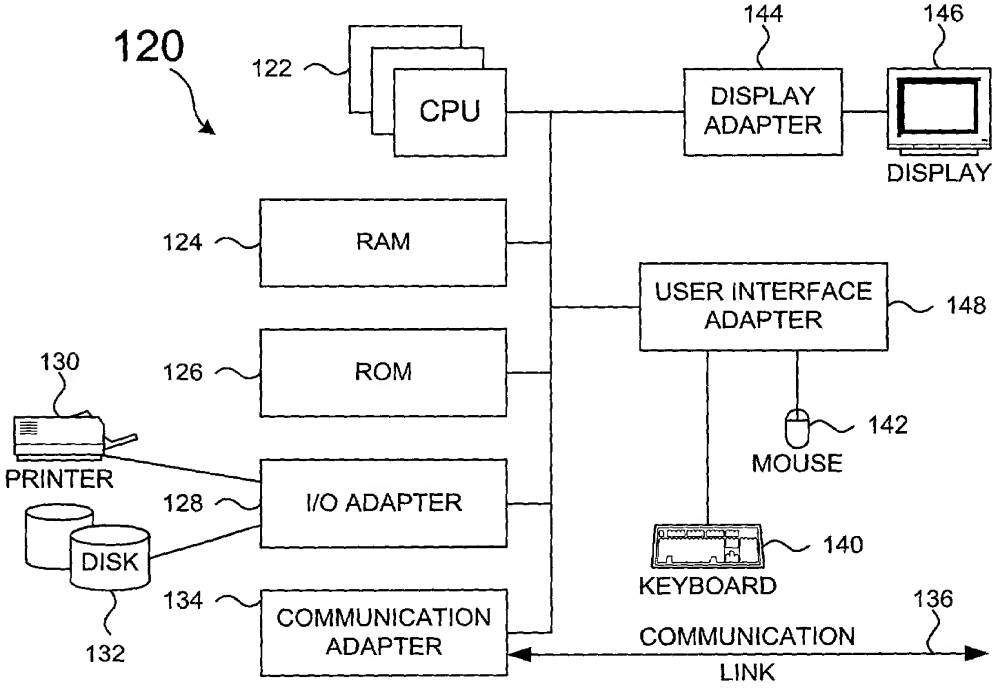

FIG. 1B
(PRIOR ART)

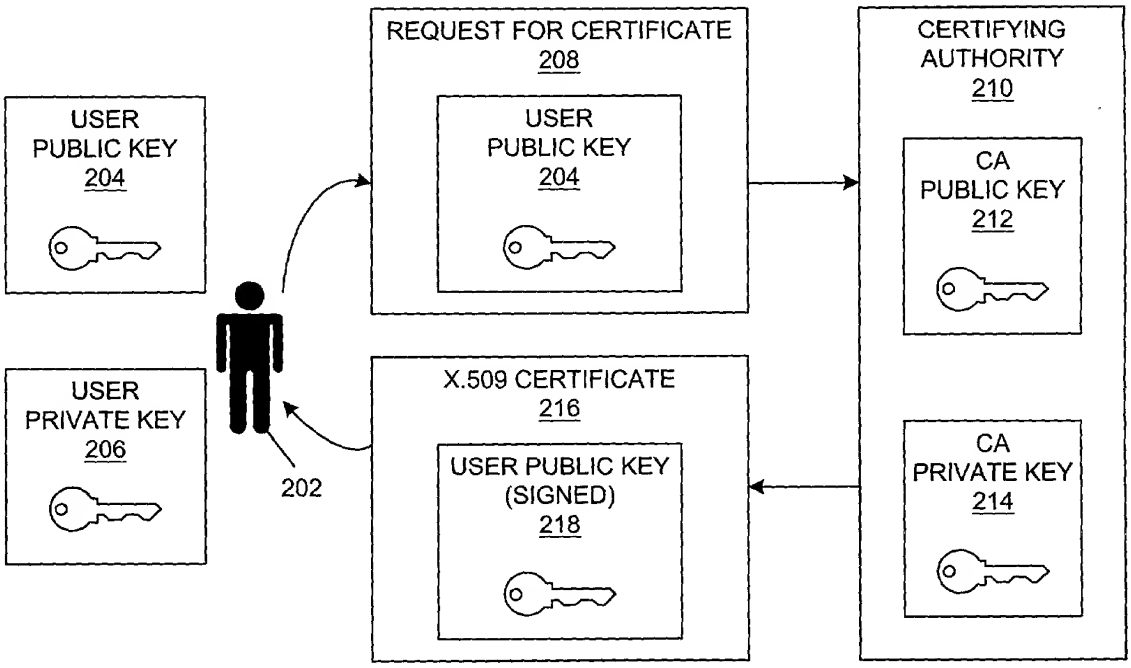

FIG. 2
(PRIOR ART)

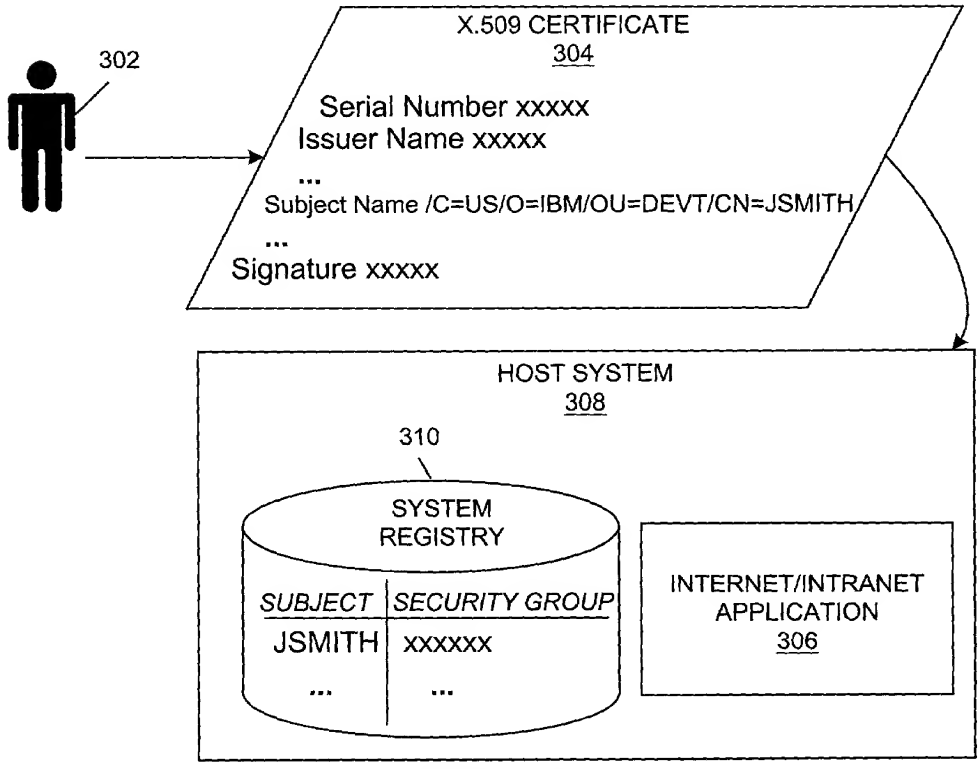

FIG. 3A
(PRIOR ART)

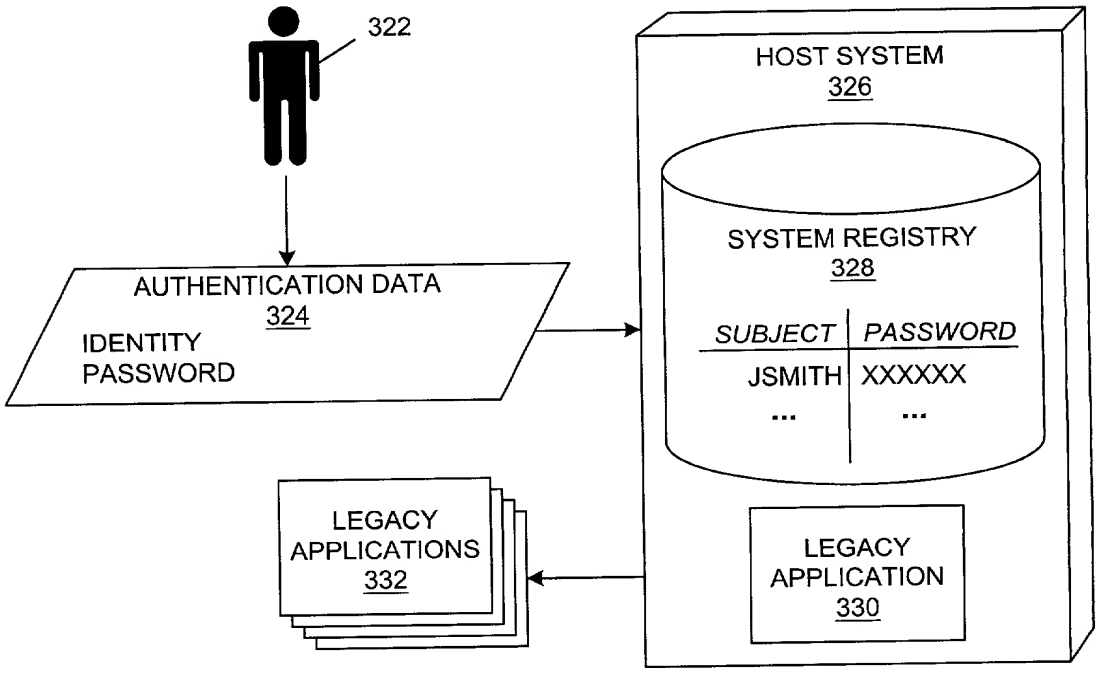

FIG. 3B
(PRIOR ART)

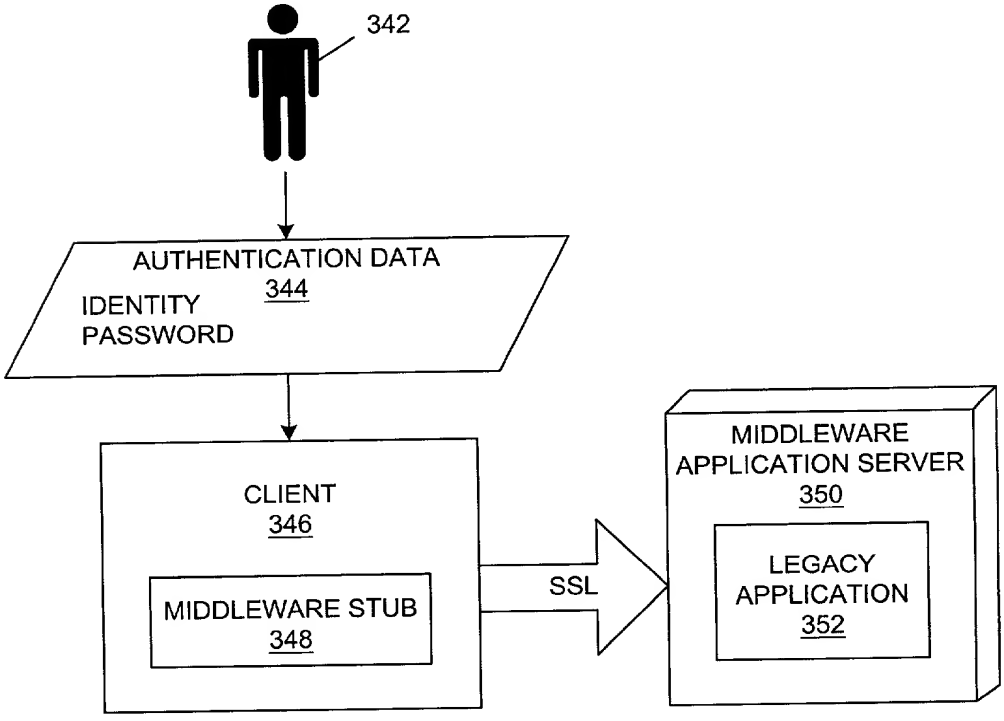

FIG. 3C
(PRIOR ART)

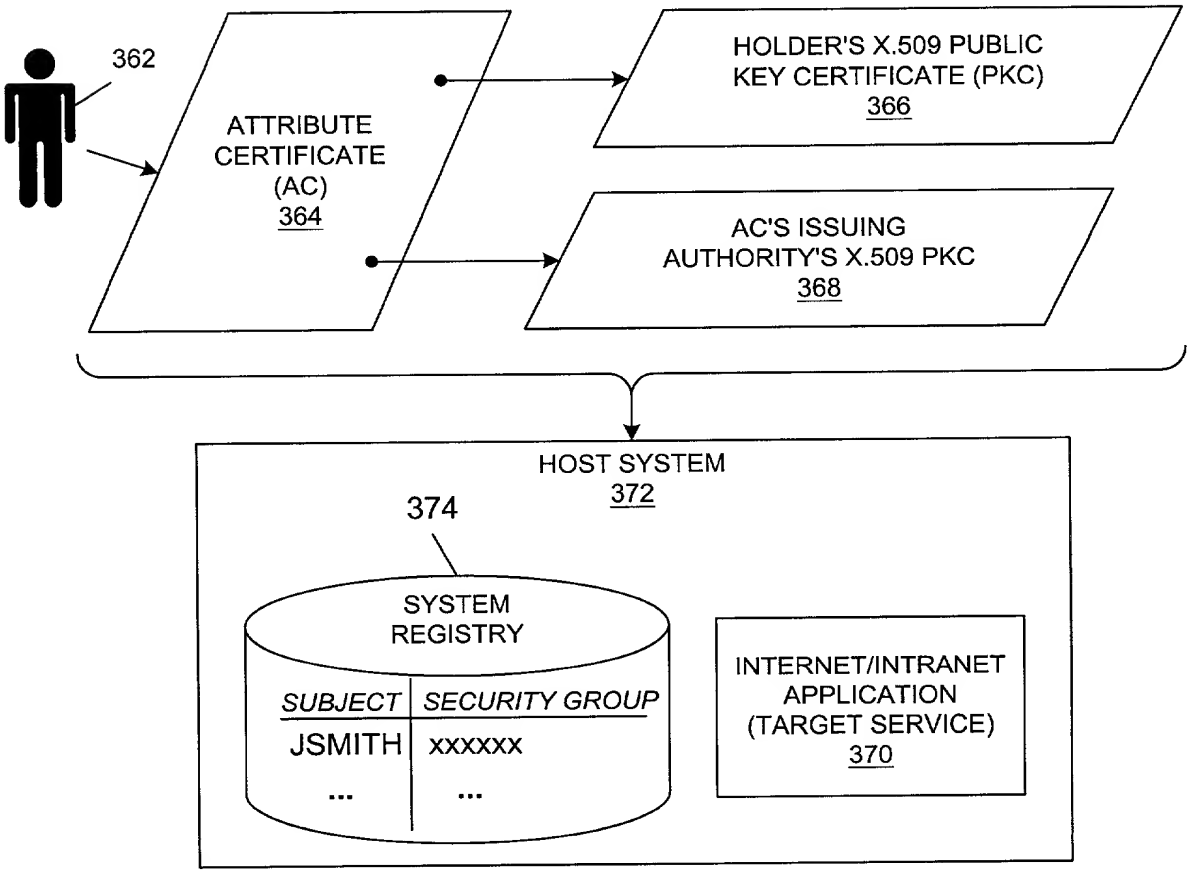

FIG. 3D
(PRIOR ART)

```
Certificate  ::=  SEQUENCE  {
      tbsCertificate           TBSCertificate,
      signatureAlgorithm       AlgorithmIdentifier,
      signature                BIT STRING  }

TBSCertificate  ::=  SEQUENCE  {
      version           [0]    Version DEFAULT v1,
      serialNumber             CertificateSerialNumber,
      signature                AlgorithmIdentifier,
      issuer                   Name,
      validity                 Validity,
      subject                  Name,
      subjectPublicKeyInfo     SubjectPublicKeyInfo,
      issuerUniqueID    [1]    IMPLICIT UniqueIdentifier OPTIONAL,
      subjectUniqueID   [2]    IMPLICIT UniqueIdentifier OPTIONAL,
      extensions        [3]    Extensions OPTIONAL   }

Version  ::=  INTEGER  {  v1(0), v2(1), v3(2)  }

CertificateSerialNumber  ::=  INTEGER

Validity ::= SEQUENCE {
      notBefore                Time,
      notAfter                 Time }

Time ::= CHOICE {
      utcTime                  UTCTime,
      generalTime              GeneralizedTime }

UniqueIdentifier  ::=  BIT STRING

SubjectPublicKeyInfo  ::=  SEQUENCE  {
      algorithm                AlgorithmIdentifier,
      subjectPublicKey         BIT STRING  }

Extensions  ::=  SEQUENCE SIZE (1..MAX) OF Extension

Extension  ::=  SEQUENCE  {
      extnID                   OBJECT IDENTIFIER,
      critical                 BOOLEAN DEFAULT FALSE,
      extnValue                OCTET STRING  }
```

FIG. 4A
(PRIOR ART)

```
AttributeCertificate ::= SEQUENCE {
      acinfo               AttributeCertificateInfo,
      signatureAlgorithm   AlgorithmIdentifier,
      signatureValue       BIT STRING
}

AttributeCertificateInfo ::= SEQUENCE {
      version                 AttCertVersion DEFAULT v1,
      holder                  Holder,
      issuer                  AttCertIssuer,
      signature               AlgorithmIdentifier,
      serialNumber            CertificateSerialNumber,
      attrCertValidityPeriod  AttCertValidityPeriod,
      attributes              SEQUENCE OF Attribute,
      issuerUniqueID          UniqueIdentifier OPTIONAL,
      extensions              Extensions OPTIONAL
}

AttCertVersion ::= INTEGER { v1(0), v2(1) }

Holder ::= SEQUENCE {
      baseCertificateID   [0] IssuerSerial OPTIONAL,
                          -- the issuer and serial number of
                          -- the holder's Public Key Certificate
      entityName          [1] GeneralNames OPTIONAL,
                          -- the name of the claimant or role
      objectDigestInfo    [2] ObjectDigestInfo OPTIONAL
                          -- if present, version must be v2
}

ObjectDigestInfo ::= SEQUENCE {
      digestedObjectType  ENUMERATED {
      publicKey            (0),
      publicKeyCert        (1),
      otherObjectTypes     (2) },
                                -- otherObjectTypes MUST NOT
                                -- be used in this profile
      otherObjectTypeID   OBJECT IDENTIFIER OPTIONAL,
      digestAlgorithm     AlgorithmIdentifier,
      objectDigest        BIT STRING
}
```

FIG. 4B
(PRIOR ART)

```
AttCertIssuer ::= CHOICE {
    v1Form                  GeneralNames,  -- v1 or v2
    v2Form          [0]  V2Form    -- v2 only
}

V2Form ::= SEQUENCE {
    issuerName              GeneralNames  OPTIONAL,
    baseCertificateID   [0]  IssuerSerial  OPTIONAL,
    objectDigestInfo    [1]  ObjectDigestInfo  OPTIONAL
                             -- at least one of issuerName, baseCertificateID
                             -- or objectDigestInfo MUST be present}

IssuerSerial  ::=  SEQUENCE {
    issuer      GeneralNames,
    serial      CertificateSerialNumber,
    issuerUID   UniqueIdentifier OPTIONAL
}

AttCertValidityPeriod  ::= SEQUENCE {
    notBeforeTime  GeneralizedTime,
    notAfterTime    GeneralizedTime
}

Attribute ::= SEQUENCE {
    type        AttributeType,
    values      SET OF AttributeValue
                -- at least one value is required
}

AttributeType ::= OBJECT IDENTIFIER

AttributeValue ::= ANY DEFINED BY AttributeType
```

FIG. 4C
(PRIOR ART)

```
name        id-aca-authenticationInfo
OID         { id-aca 1 }
Syntax      SvceAuthInfo
values:     Multiple allowed

SvceAuthInfo ::=   SEQUENCE {
    service     GeneralName,
    ident       GeneralName,
    authInfo    OCTET STRING OPTIONAL
}
```

FIG. 4D
(PRIOR ART)

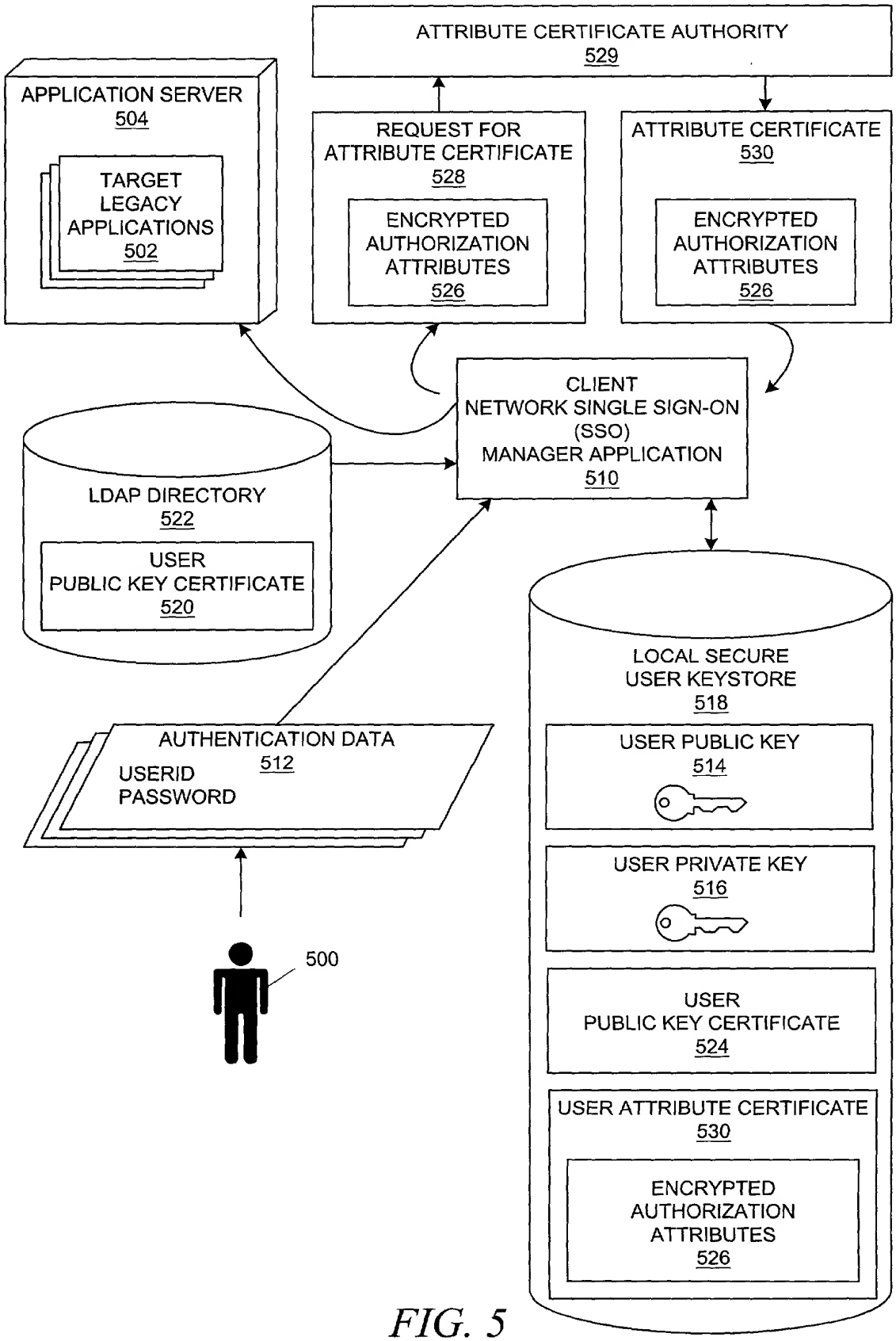
ATTRIBUTE CERTIFICATE AUTHORITY
529
APPLICATION SERVER
504
TARGET LEGACY APPLICATIONS
502
REQUEST FOR ATTRIBUTE CERTIFICATE
528
ENCRYPTED AUTHORIZATION ATTRIBUTES
526
ATTRIBUTE CERTIFICATE
530
ENCRYPTED AUTHORIZATION ATTRIBUTES
526
CLIENT NETWORK SINGLE SIGN-ON (SSO) MANAGER APPLICATION
510
LDAP DIRECTORY
522
USER PUBLIC KEY CERTIFICATE
520
LOCAL SECURE USER KEYSTORE
518
USER PUBLIC KEY
514
USER PRIVATE KEY
516
USER PUBLIC KEY CERTIFICATE
524
USER ATTRIBUTE CERTIFICATE
530
ENCRYPTED AUTHORIZATION ATTRIBUTES
526
AUTHENTICATION DATA
512
USERID
PASSWORD
500

FIG. 5

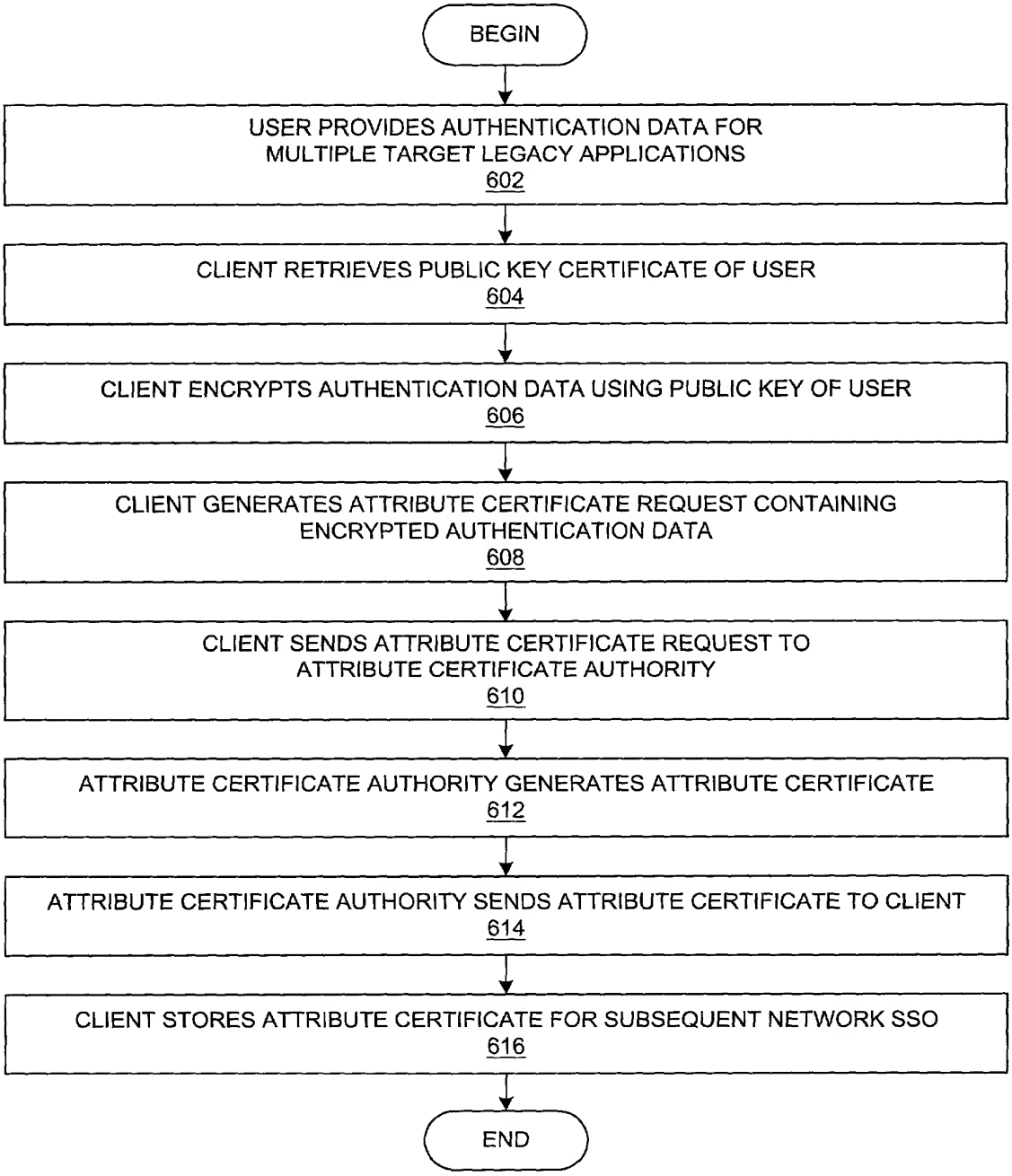
BEGIN
USER PROVIDES AUTHENTICATION DATA FOR MULTIPLE TARGET LEGACY APPLICATIONS
602
CLIENT RETRIEVES PUBLIC KEY CERTIFICATE OF USER
604
CLIENT ENCRYPTS AUTHENTICATION DATA USING PUBLIC KEY OF USER
606
CLIENT GENERATES ATTRIBUTE CERTIFICATE REQUEST CONTAINING ENCRYPTED AUTHENTICATION DATA
608
CLIENT SENDS ATTRIBUTE CERTIFICATE REQUEST TO ATTRIBUTE CERTIFICATE AUTHORITY
610
ATTRIBUTE CERTIFICATE AUTHORITY GENERATES ATTRIBUTE CERTIFICATE
612
ATTRIBUTE CERTIFICATE AUTHORITY SENDS ATTRIBUTE CERTIFICATE TO CLIENT
614
CLIENT STORES ATTRIBUTE CERTIFICATE FOR SUBSEQUENT NETWORK SSO
616
END

FIG. 6

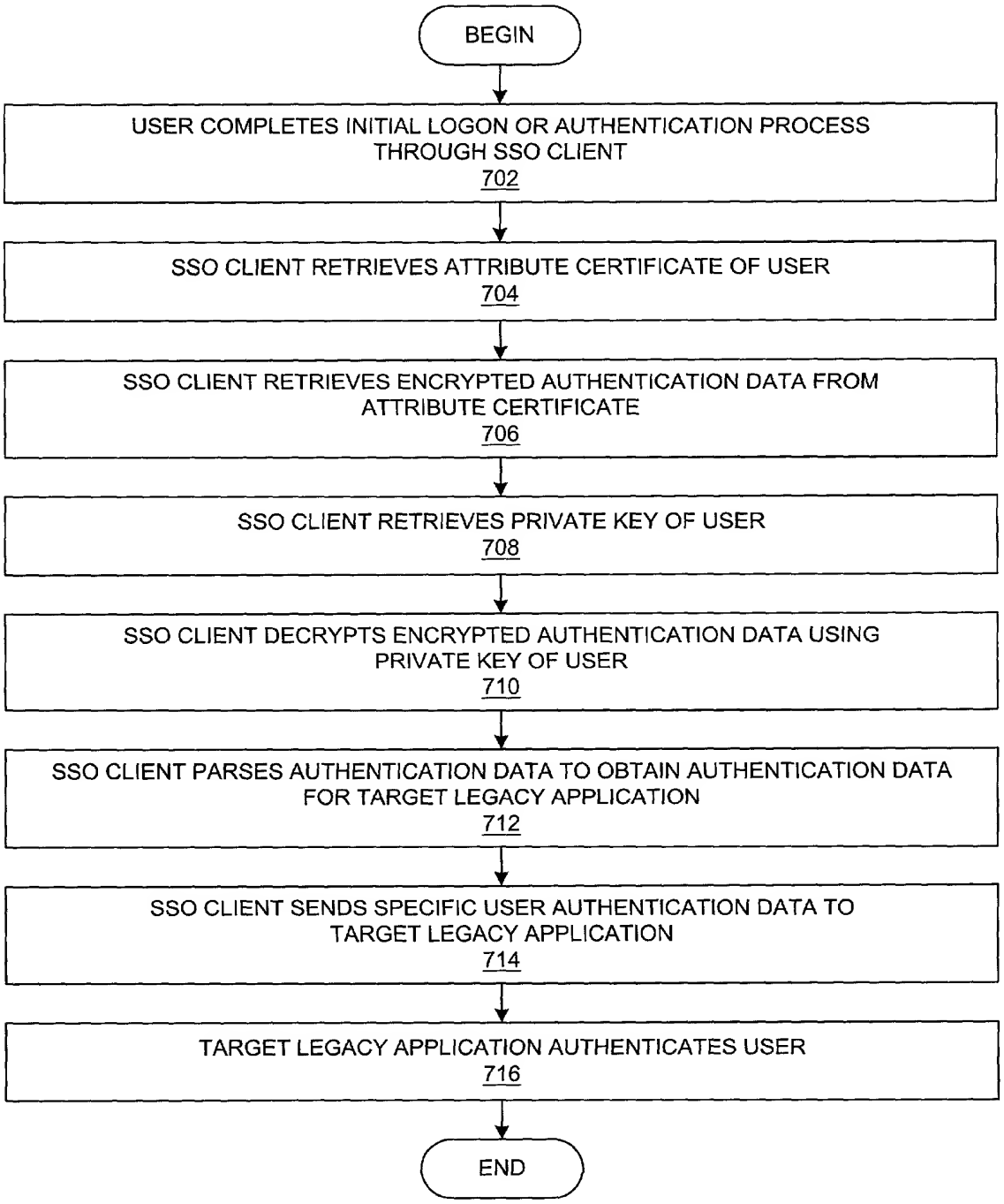
BEGIN
USER COMPLETES INITIAL LOGON OR AUTHENTICATION PROCESS THROUGH SSO CLIENT
702
SSO CLIENT RETRIEVES ATTRIBUTE CERTIFICATE OF USER
704
SSO CLIENT RETRIEVES ENCRYPTED AUTHENTICATION DATA FROM ATTRIBUTE CERTIFICATE
706
SSO CLIENT RETRIEVES PRIVATE KEY OF USER
708
SSO CLIENT DECRYPTS ENCRYPTED AUTHENTICATION DATA USING PRIVATE KEY OF USER
710
SSO CLIENT PARSES AUTHENTICATION DATA TO OBTAIN AUTHENTICATION DATA FOR TARGET LEGACY APPLICATION
712
SSO CLIENT SENDS SPECIFIC USER AUTHENTICATION DATA TO TARGET LEGACY APPLICATION
714
TARGET LEGACY APPLICATION AUTHENTICATES USER
716
END

FIG. 7